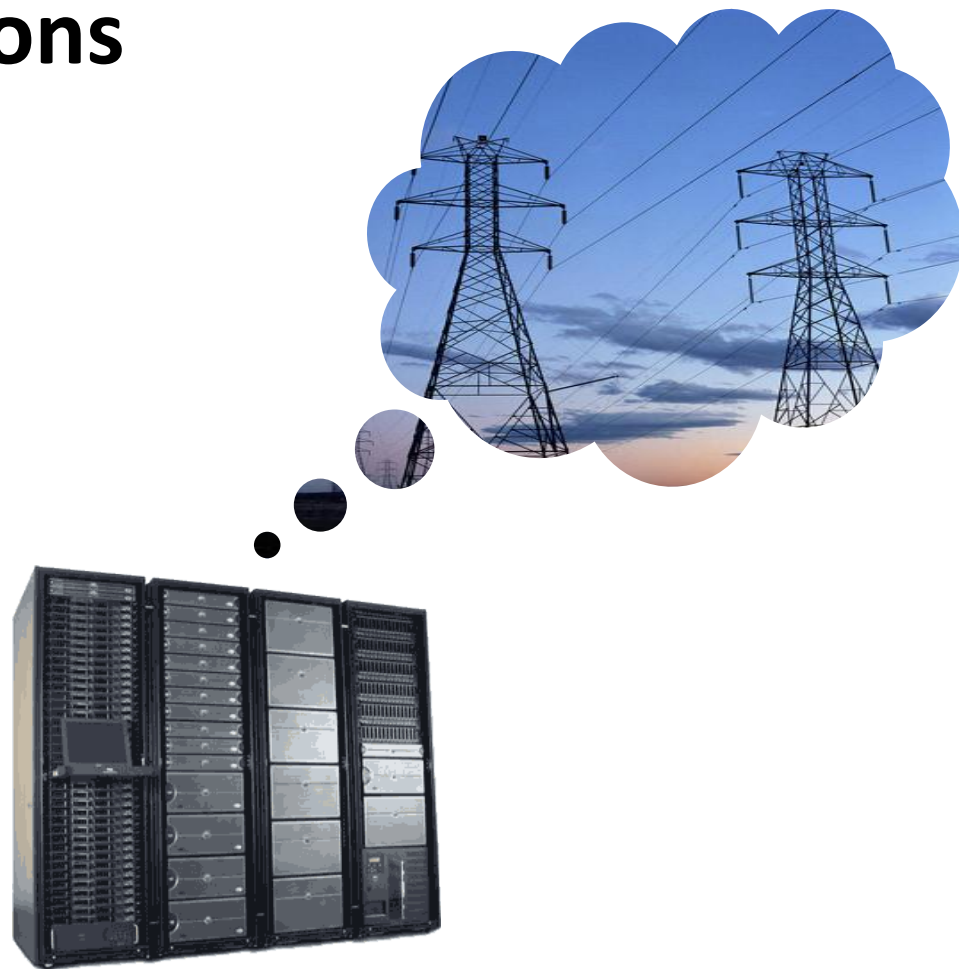


# Trends in High-Performance Computing for Power Grid Applications

**Franz Franchetti**

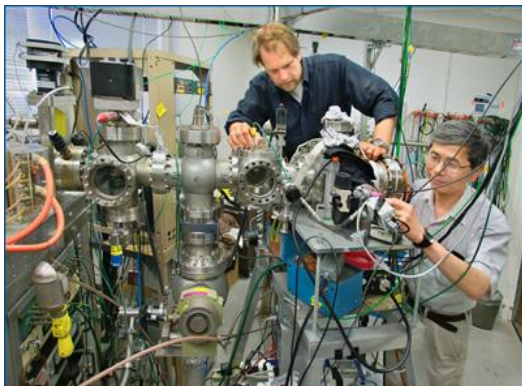
ECE, Carnegie Mellon University  
[www.spiral.net](http://www.spiral.net)

Co-Founder, SpiralGen  
[www.spiralgen.com](http://www.spiralgen.com)



This talk presents my personal views and is not endorsed by any other party mentioned in it. The copyright of all images is held by the respective owners. They are used under “fair use.”

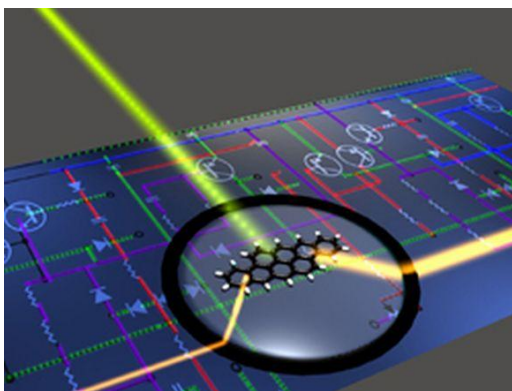
# This Talk Is Not About 2050...



**Quantum computing**



**DNA computing**



**Optical computing**



**Other far-out technologies**

# This Talk: HPC and Supercomputing in 2018



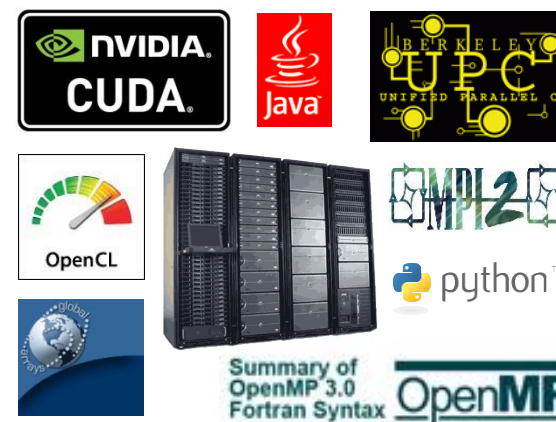
Where are we today?



How did we end up here?



Where will we go?



How will it impact you?

# What Is High Performance Computing?

1 flop/s = one floating-point operation (addition or multiplication) per second

mega (M) =  $10^6$ , giga (G) =  $10^9$ , tera (T) =  $10^{12}$ , peta (P) =  $10^{15}$ , exa (E) =  $10^{18}$

## Computing systems in 2010



### Cell phone

1 CPUs  
1 Gflop/s  
\$300  
1 W power



### Laptop

2 CPUs  
20 Gflop/s  
\$1,200  
30 W power



### Workstation

8 CPUs  
1 Tflop/s  
\$10,000  
1 kW power



### HPC

200 CPUs  
20 Tflop/s  
\$700,000  
8 kW power



### #1 supercomputer

224,162 CPUs  
2.3 Pflop/s  
\$100,000,000  
7 MW power



### Power grid scenario

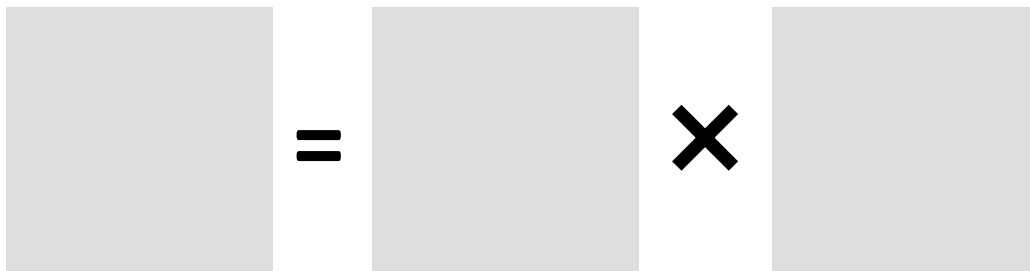
- Central servers (planning, contingency analysis)
- Autonomous controllers (smart grids)
- Operator workstations (decision support)

# How Big are the Computational Problems?

1 flop/s = one floating-point operation (addition or multiplication) per second

mega (M) =  $10^6$ , giga (G) =  $10^9$ , tera (T) =  $10^{12}$ , peta (P) =  $10^{15}$ , exa (E) =  $10^{18}$

## Matrix-matrix multiplication...



```
for i=1:n
  for j=1:n
    for k=1:n
      C[i,j] =
        A[i,k]*B[k,j]
```

## ..running on...



### Cell phone

1 Gflop/s

**1k × 1k**

**8MB, 2s**



### Laptop

20 Gflop/s

**8k × 8k**

**0.5 GB, 5.5s**



### Workstation

1 Tflop/s

**16k × 16k**

**2 GB, 8s**



### HPC

20 Tflop/s

**64k × 64k**

**32 GB, 28s**



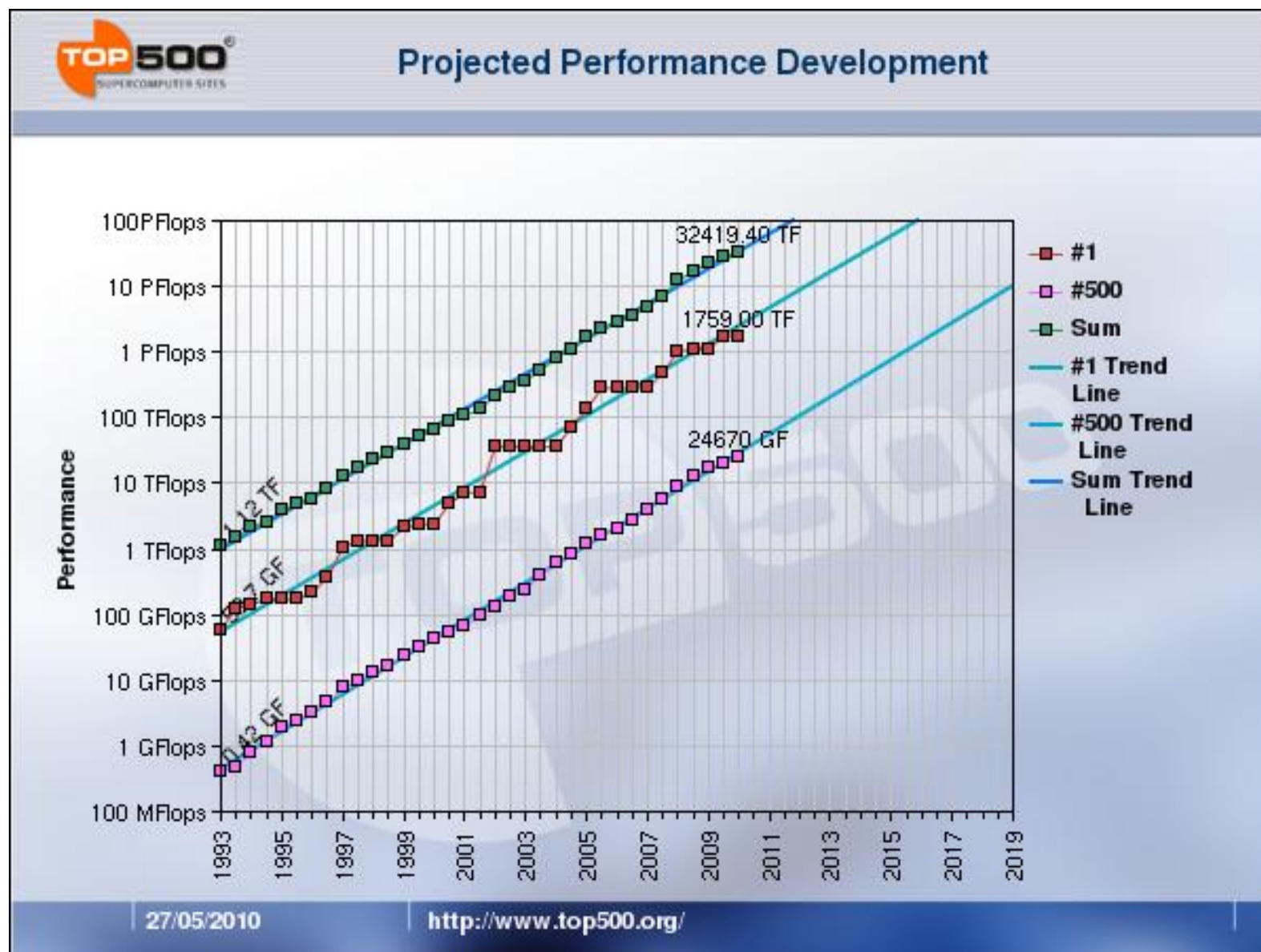
### #1 supercomputer

2.3 Pflop/s

**1M × 1M**

**8 TB, 1,000s**

# The Evolution of Performance



# How Do We Compare?

1 flop/s = one floating-point operation (addition or multiplication) per second

mega (M) =  $10^6$ , giga (G) =  $10^9$ , tera (T) =  $10^{12}$ , peta (P) =  $10^{15}$ , exa (E) =  $10^{18}$

**In 2010...**



**Cell phone**  
1 Gflop/s



**Laptop**  
20 Gflop/s



**Workstation**  
1 Tflop/s



**HPC**  
20 Tflop/s



**#1 supercomputer**  
2.3 Pflop/s

**...would have been the #1 supercomputer back in...**



**Cray X-MP/48**  
941 Mflop/s  
**1984**



**NEC SX-3/44R**  
23.2 Gflop/s  
**1990**



**Intel ASCI Red**  
1.338 Tflop/s  
**1997**



**Earth Simulator**  
35.86 Tflop/s  
**2002**

# If History Predicted the Future...

...the performance of the #1 supercomputer of 2010...



**#1 supercomputer**  
1 Pflop/s

...could be available as



**HPC**  
1 Pflop/s  
**2018**



**Workstation**  
1 Pflop/s  
**2023**



**Laptop**  
1 Pflop/s  
**2030**



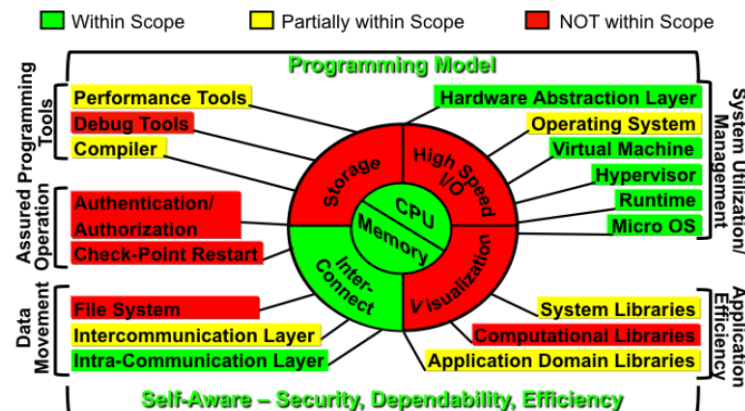
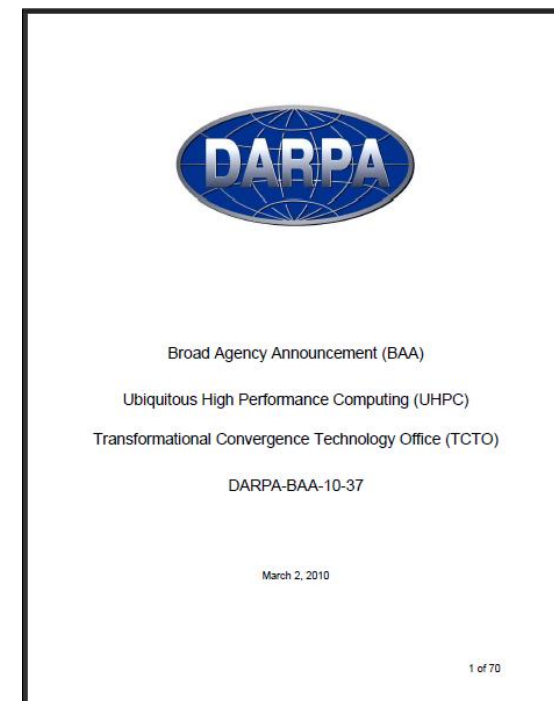
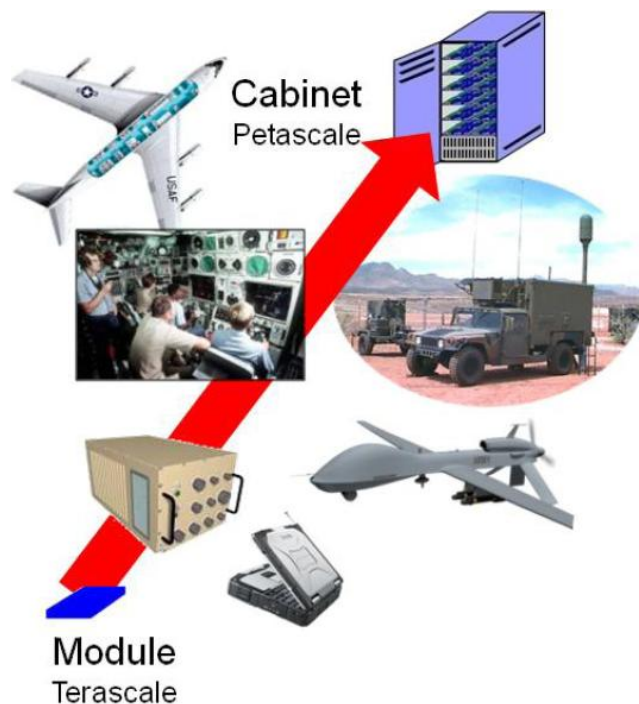
**Cell phone**  
1 Pflop/s  
**2036**

How do we get here?

# HPC: ExtremeScale Computing

## DARPA UHPC ExtremeScale system goals

- Time-frame: 2018
- 1 Pflop/s, air-cooled, single 19-inch cabinet
- Power budget: 57 kW, including cooling
- 50 Gflop/W for HPL benchmark



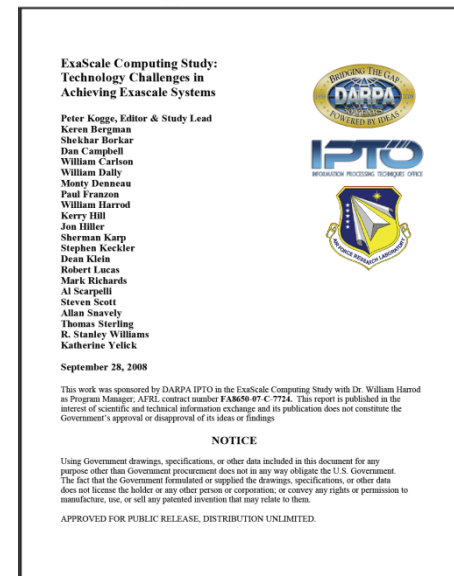
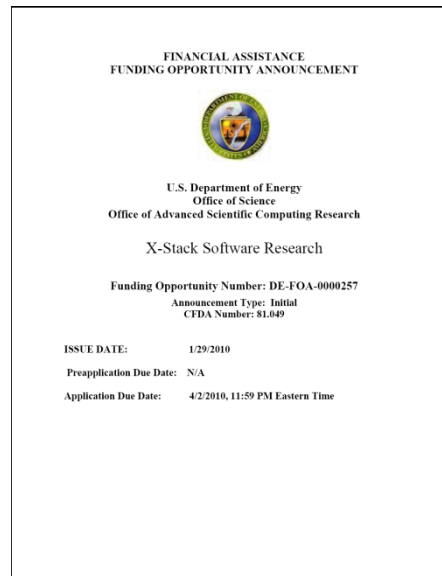
# Developing The New #1: DoE ExaScale

## Challenges to achieve ExaScale

- Energy and power
- Memory and storage
- Concurrency and locality
- Resiliency

## X-Stack: Software for ExaScale

- System software
- Fault management
- Programming environments
- Applications frameworks
- Workflow systems



**#1 supercomputer**  
2 Pflop/s  
**2010**



**#1 supercomputer**  
1 Eflop/s  
**2018**

# Some Predictions for ExaScale Machines

## Processors

- 10 billion-way concurrency
- 100's of cores per die
- 10 to 10-way per-core concurrency
- 100 million to 1 billion cores at 1 to 2 GHz
- Multi-threaded fine grain concurrency
- 10,000s of cycles system-wide latency

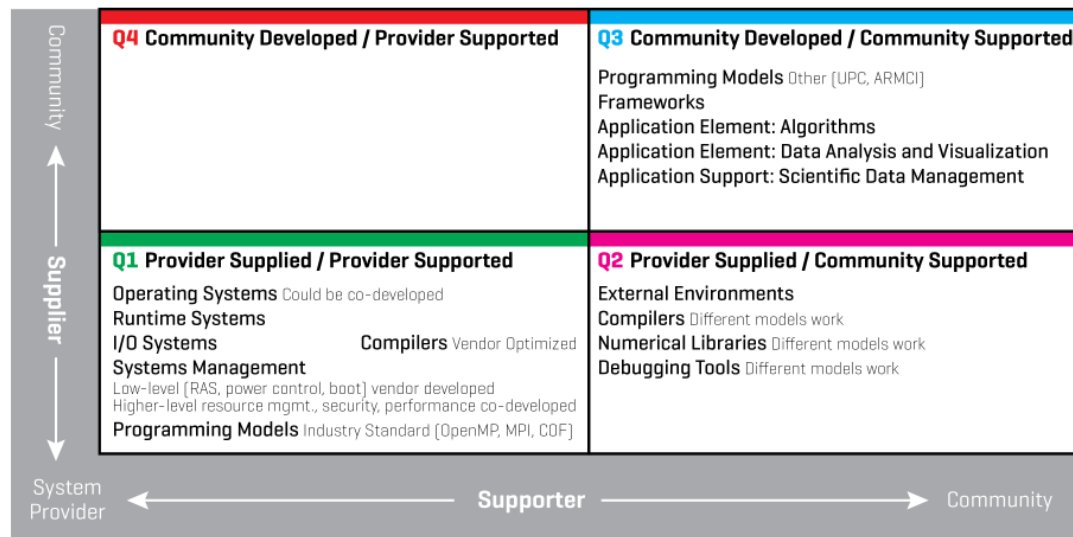


## Memory

- Global address space without cache coherence
- Explicitly managed high speed buffer caches
- 128 PB capacity
- Deep memory hierarchies

## Technology

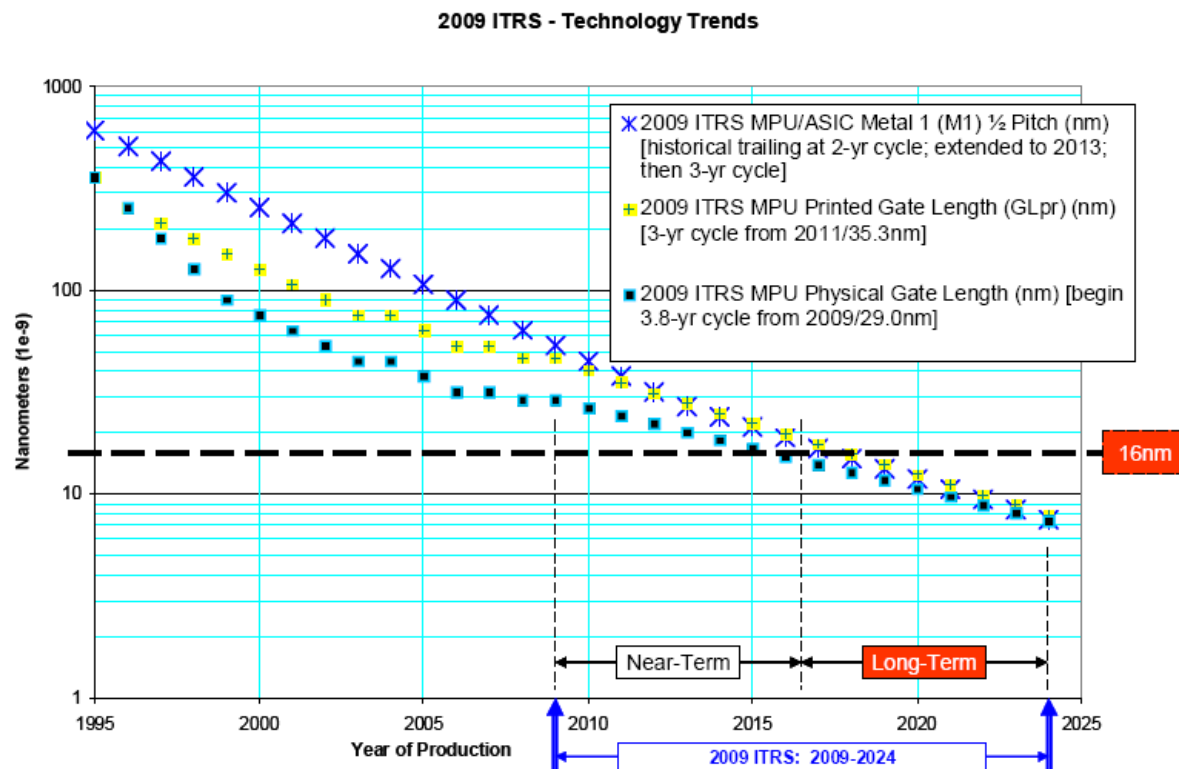
- 22 to 11 nanometers CMOS
- 3-D packaging of dies
- Optical communications at 1TB/s
- Fault tolerance
- Active power management



# International Semiconductor Roadmap

## Near-term (through 2016) and long-term (2017 through 2024)

- Process Integration, Devices, and Structures
- RF and Analog / Mixed-signal Technologies for Wireless Communications
- Emerging Research Devices
- System Drivers
- Design
- Test and Test Equipment
- Front End Processes
- Lithography
- Interconnect
- Factory Integration
- Assembly and Packaging



International Technology Roadmap for Semiconductors

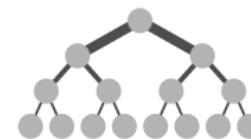
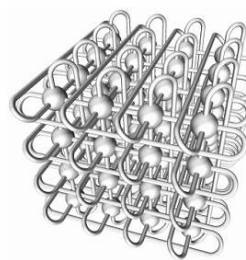
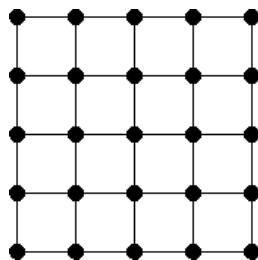
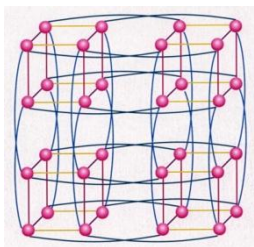
# Prediction 1: Network of Nodes

## Why?

- State-of-the-art for large machines
- Allows scaling from Tflop/s to Eflop/s
- Designs can be tailored to application
- Fault tolerance

## Implications

- Segmented address space
- Multiple instructions, multiple data (MIMD)
- Packet-based messaging
- Long inter-node latencies



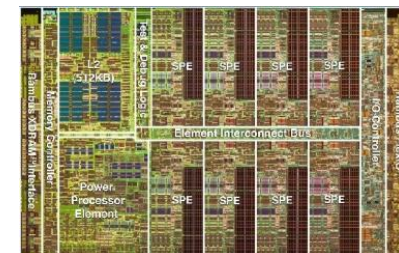
# Prediction 2: Multicore CPUs

## Why?

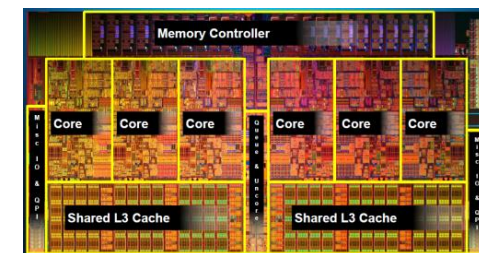
- State-of-the-art CPU design
- Growing transistor count (Moore's law)
- Limited power budget

## Implications

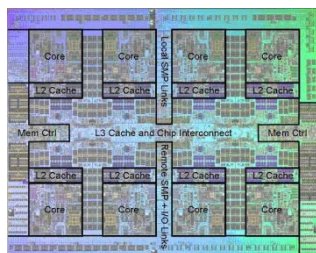
- On-chip multithreading
- Instruction set extensions targeting applications
- Physically segmented cache
- Software and/or hardware managed cache
- Non-uniform memory access (NUMA)



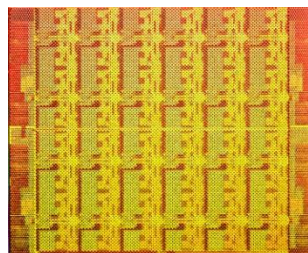
**IBM Cell BE**  
8+1 cores



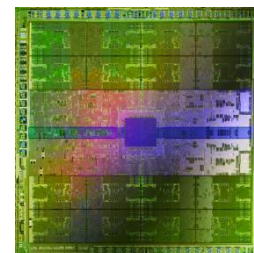
**Intel Core i7**  
8 cores, 2-way SMT



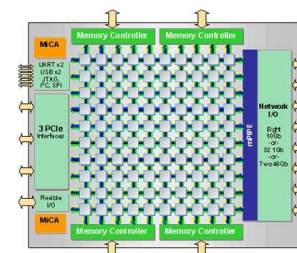
**IBM POWER7**  
8 cores, 4-way SMT



**Intel SCC**  
48 cores



**Nvidia Fermi**  
448 cores, SMT



**Tiler TILE Gx**  
100 cores

# Prediction 3: Accelerators

## Why?

- Special purpose enables better efficiency
- 10x to 100x gain for data parallel problems
- Limited applicability, thus co-processor
- Can be discrete chip or integrated on die

## Implications

- Multiple programming models
- Coarse-grain partitioning necessary
- Programs often become non-portable



### Rack-mount server components

2 quad-core CPUs + 4 GPUs  
200 Gflop/s + 4 Tflop/s



### RoadRunner

6,480 CPUs + 12,960 Cells  
3240 TriBlades



### HPC cabinet

CPU blades + GPU blades  
Custom interconnect

# Prediction 4: Memory Capacity Limited

## Why?

- Good machine balance: 1 byte/flop
- Multicore CPUs have huge performance
- Limited power budget
- Need to limit memory size

## Implications

- Saving memory complicates programs
- Trade-off: memory vs. operations
- Requires new algorithm optimization



### Dell PowerEdge R910

2 x 8-core CPUs  
256 GB, 145 Gflop/s  
1 core: 16 GB for 9 Gflop/s  
**1.7 byte/flop**



### BlueGene/L

65,536 dual-core CPUs  
16 TB RAM, 360 Tflop/s  
1 core: 128 MB for 2.8 Gflop/s  
**0.045 byte/flop**



### Nvidia Tesla M2050 (Fermi)

1 GPU, 448 cores  
6 GB, 515 Gflop/s  
1 core: 13 MB for 1.15 Gflop/s  
**0.011 byte/flop**

# HPC Software Development

## Popular HPC programming languages

- 1953: Fortran
- 1973: C
- 1985: C++
- 1997: OpenMP
- 2007: CUDA

## Popular HPC libraries

- 1979: BLAS
- 1992: LAPACK
- 1994: MPI
- 1995: ScaLAPACK
- 1995: PETSc
- 1997: FFTW

## Proposed and maturing (?)

- Chapel, X10, Fortress, UPC, GA, HTA, OpenCL, Brook, Sequoia, Charm++, CnC, STAPL, TBB, Cilk,...

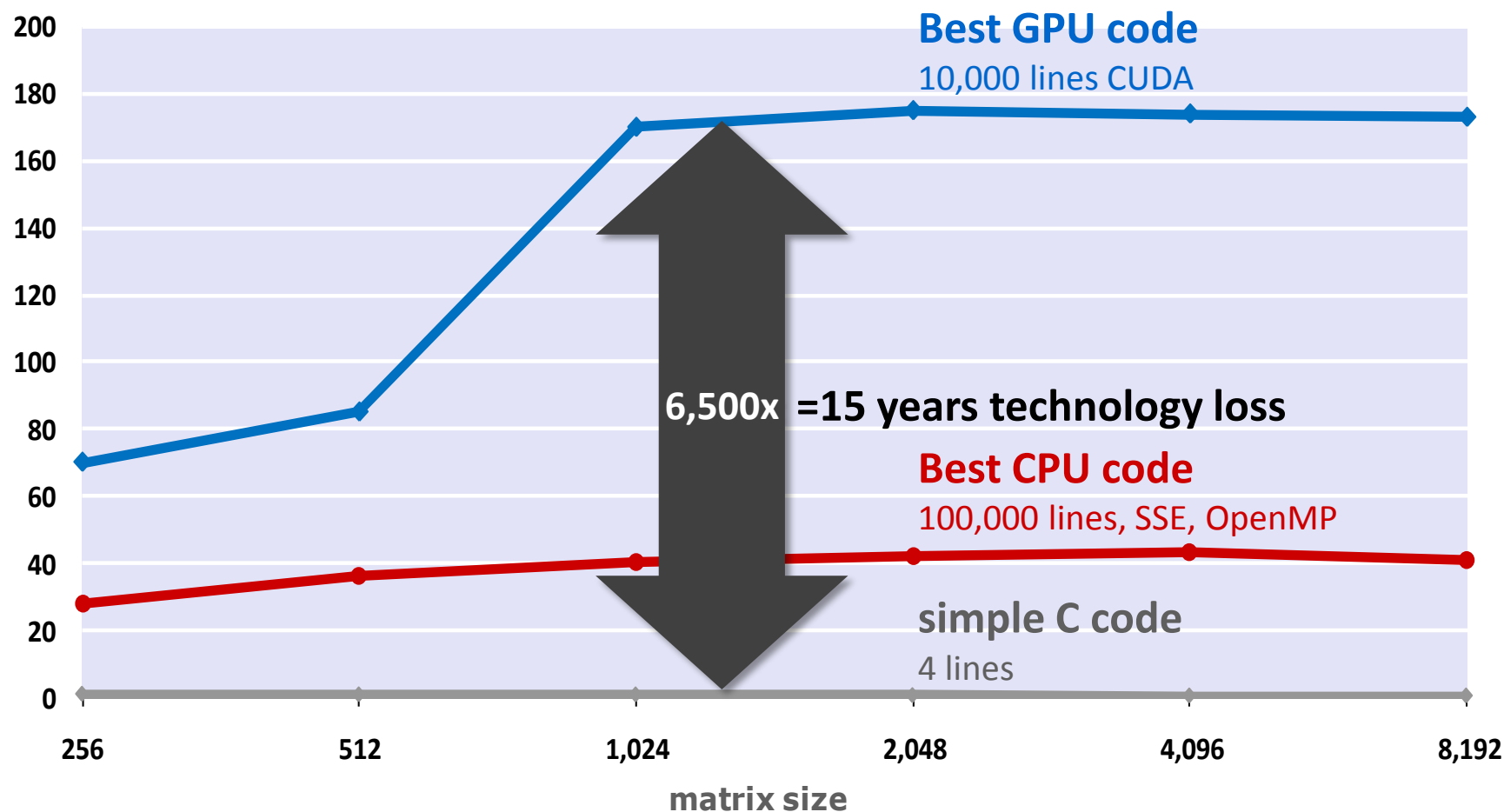


**Slow change in direction**

# The Cost Of Portability and Maintainability

## Matrix-Matrix Multiplication

Performance [Gflop/s]



# Summary

- Hardware vendors will somehow keep Moore's law on track

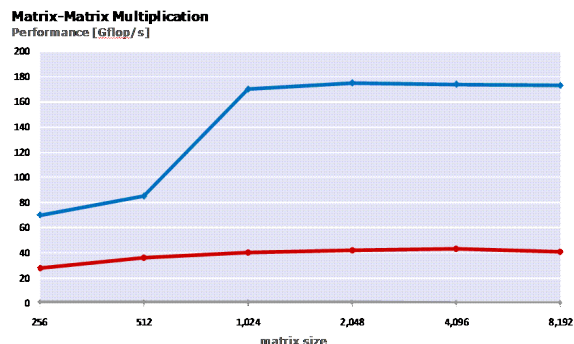


U.S. Department of Energy  
Office of Science  
Office of Advanced Scientific Computing Research

- Software development changes very slowly



- Portable and maintainable code costs performance



Unoptimized program  
= 15 years technology loss